Bolt Beranek and Newman Inc.

(12)

Report No. 5242

Requirements for Natural Language Understanding in a System with Graphic Displays

Candace L. Sidner and Madeleine Bates

March 1983

Prepared for
Defense Advanced Research Projects Agency

Unclassified

SECURITY CLASSIFICATION OF THIS PAGE (When Data Entered)

REPORT DOCUMENTATION PAGE — READ INSTRUCTIONS BEFORE COMPLETING FORM

1. REPORT NUMBER: BBN Report No. 5242
2. GOVT ACCESSION NO.: AD-A126065
3. RECIPIENT'S CATALOG NUMBER:
4. TITLE (and Subtitle): REQUIREMENTS FOR NATURAL LANGUAGE UNDERSTANDING IN A SYSTEM WITH GRAPHIC DISPLAYS
5. TYPE OF REPORT & PERIOD COVERED: Technical Report
6. PERFORMING ORG. REPORT NUMBER: BBN ~~Report No.~~ 5242
7. AUTHOR(s): Candace L. Sidner and Madeleine Bates
8. CONTRACT OR GRANT NUMBER(s): N00014-77-C-0378
9. PERFORMING ORGANIZATION NAME AND ADDRESS: Bolt Beranek and Newman Inc.
10 Moulton Street
Cambridge, MA 02238
10. PROGRAM ELEMENT, PROJECT, TASK AREA & WORK UNIT NUMBERS:
11. CONTROLLING OFFICE NAME AND ADDRESS: Office of Naval Research
Department of the Navy
Arlington, VA 22217
12. REPORT DATE: March 1983
13. NUMBER OF PAGES: 49
14. MONITORING AGENCY NAME & ADDRESS (if different from Controlling Office):
15. SECURITY CLASS. (of this report): Unclassified
15a. DECLASSIFICATION/DOWNGRADING SCHEDULE:
16. DISTRIBUTION STATEMENT (of this Report): Distribution of this document is unlimited. It may be released to the Clearing house, Department of Commerce, for sale to the general public.
17. DISTRIBUTION STATEMENT (of the abstract entered in Block 20, if different from Report):
18. SUPPLEMENTARY NOTES:
19. KEY WORDS (Continue on reverse side if necessary and identify by block number): Natural Language; KL-ONE; graphics; syntax; parsing; semantics; pragmatics; speaker intention; knowledge representation
20. ABSTRACT (Continue on reverse side if necessary and identify by block number):

Many research and applications groups are attempting to develop natural language interfaces to systems of many different types. The domain being used, the degree of "intelligence" the system should exhibit, and other characteristics greatly affect the way the language capability should be designed. However

DD FORM 1473, 1 JAN 73 — EDITION OF 1 NOV 65 IS OBSOLETE

Unclassified

SECURITY CLASSIFICATION OF THIS PAGE (When Data Entered) there is no generally accepted (or even commonly used) method of determining, in the early stages of the design process, just what capacities the particular natural language interface must possess in order to be effective. In this paper we set forth a case study of a methodology that has been extremely effective in our domain and which can easily be adapted to other situations.

The particular task we explored was that of a decision maker examining and modifying a database using a graphics display. The decision maker is expected to manipulate both the context of the database and the form of the display. The system is expected to be a helpful, intelligent assistant with considerable linguistic capability so that the user can express commands, questions, facts, and other material very naturally. The system is based on an analysis of protocols obtained from users interacting with a simulated graphics and natural language system.

This paper presents three aspects of our research on a system that can provide graphically represented information and can talk naturally with a user about that information:

1. description of the methodology that we used in developing and analyzing an extended prototypical dialogue between a user and such a system;

2. portions of our analysis of that dialogue that present both the information obtained and the method of obtaining it; and

3. conclusions about the necessary linguistic and non-linguistic capacities of an intelligent conversational partner, as drawn from the full analysis.

Report No. 5242

REQUIREMENTS FOR NATURAL LANGUAGE UNDERSTANDING IN A SYSTEM WITH GRAPHIC DISPLAYS

Candace L. Sidner and Madeleine Bates

March 1983

Prepared for:

Defense Advanced Research Projects Agency
1400 Wilson Boulevard
Arlington, VA 22209

ARPA Order No. 3414 Contract No. N00014-77-C-0378

Effective Date of Contract: 1 September 1977
Contract Expiration Date: 30 September 1984

This research was supported by the Advanced Research Projects Agency of the Department of Defense and was monitored by ONR under Contract No. N00014-77-C-0378. The views and conclusions contained in this document are those of the authors and should not be interpreted as necessarily representing the official policies, either expressed or implied, of the Defense Advanced Research Projects Agency or the U.S. Government.

TABLE OF CONTENTS

	Page
1 Introduction	1
2 Methodology and Construction of the Scenario	3
3 Observations on the Scenario	12
3.1 Exchange 1	12
3.2 Exchange 2	15
3.3 Exchange 3	19
3.4 Exchange 4	22
3.5 Exchange 5	24
3.6 Exchange 6	26
4 Conclusions	27
APPENDIX A. SEMANTIC INTERPRETATION OF UTTERANCES [1A]-[6A] USING KL-ONE	33
Figure 1	34
Figure 2	35
Figure 3	36
Figure 4	37
Figure 5	38
Figure 6	39
Figure 7	40
Figure 8	41
Figure 9	42
Figure 10	43
Figure 11	44
Figure 12	45
Figure 13	46
Figure 14	47

1 Introduction

Many research and applications groups are attempting to develop natural language interfaces to systems of many different types. The domain being used, the degree of "intelligence" the system should exhibit, and other characteristics greatly affect the way the language capability should be designed. However there is no generally accepted (or even commonly used) method of determining, in the early stages of the design process, just what capacities the particular natural language interface must possess in order to be effective. In this paper we[1] set forth a case study of a methodology that has been extremely effective in our domain and which can easily be adapted to other situations.

The Knowledge Representation and Natural Language group at BBN is building natural language tools to help decision makers explore and modify a database using a graphics display. These decision makers will manipulate both the context of the database and the form of the display. We want our users to be able to express themselves very naturally, that is, to be able to utter more than direct imperatives and to ask questions other than the direct questions typical of most current AI language systems. One important feature of the system is that it permits non-verbal behavior on the part of both the system and the user; the user can point at the screen and the system can respond to the user by changing the display as well as producing textual responses.

---

[1]We wish to acknowledge the contributions of our colleagues Rusty Bobrow and David Israel in the research that is reported here.

To help us understand the special issues of language processing in this environment, we collected protocols, by a method to be described later, of users interacting with simulated versions of the system we envision. Our analysis of those protocols convinced us that we needed a system with very different kinds of linguistic capabilities than are required in environments without graphics or with restrictions on the kind of utterances the user may produce. In our protocols, users ask direct information retrieval questions, report errors, request action via questions, include several utterances in one request, ask elliptical questions, seek clarifications, and report plan changes and new plans. This means that the system must at least have the capacity to interpret many different kinds of referential and deictic phrases, to uncover the intended meanings of these sentence types, and to recognize errors in the user's plans.

This paper presents three aspects of our research on a system that can provide graphically represented information and can talk naturally with a user about that information:

1. a description of the methodology that we used in developing and analyzing an extended prototypical dialogue between a user and such a system,

2. portions of our analysis of that dialogue in which we present both the information obtained and the method of obtaining it, and

3. conclusions about the necessary linguistic and non-linguistic capacities of an intelligent conversational partner, as drawn from the full analysis.

The methodology of scenario development and analysis can be generalized to systems without graphic capabilities and with a lower degree of cooperative behavior; many of the conclusions presented are also relevant to such systems. Most of the particular capabilities determined by this case study are necessary for any automated cooperative conversational partner.

2 Methodology and Construction of the Scenario

What can be learned from studying an extended dialogue between a simulated natural language understanding system and someone using that system to perform a moderately complex task? Such a dialogue provides information that cannot be obtained without studying some realistic example in detail.

- In a realistic dialogue with a large number of exchanges, users formulate and carry out extended plans, make changes in those plans and alter their behavior based upon the feedback they get from the system. Protocols provide a record of these dialogues.

- Protocols are be viewed and studied as a trace of a particular path through a very large dialogue space; each utterance is the result of a procedure that must not only eliminate inappropriate utterances but also choose (or generate) one of fairly small set of "best" utterances.

- The protocols are a trace of user-system information flow. The flow of communication between user and system cannot be gleaned from introspective speculation since in real communication the partners do not have full knowledge of one another's state of mind. In introspection one cannot realistically pretend not to know the state of one's mind (just as most two-person games cannot be played realistically by one person!).

- By using an intelligent partner in the conversation, as opposed to a dumb machine, the user can rely on his/her partner for problem solving capabilities both as a language user and as a task assistant.

Our methodology can be summarized as a four step process:

1. collect protocols of appropriate tasks for domains of interest,

2. analyze these protocols for the purpose of choosing a scenario of possible behaviors that are exhibited in the protocols and could be handled by research tools and theories available or under development,

3. analyze the scenario for an initial system design,

4. refine the system design and repeat the scenario analysis for further design constraints.

We have pursued this methodology for the system design reported in this paper. In particular, since we were studying communication in cooperative tasks with graphics displays, we chose to take protocols in which a user "talked" with a hypothetical system about tasks that involved representing objects graphically. We chose tasks that demanded that each participant have some knowledge of the other, but that limited their knowledge enough that each would have to ask the other for some information. We also wanted to gain some insight into people's deictic behavior and so we constructed the tasks and the tools to allow for this possibility (which people used to varying degrees).

The scenario presented later is based upon fragments of several dialogues between a user and a simulated system for the language and graphics world, which we call KLONE-ED [9]. These dialogues were obtained by having one person play the role of the system and the other person use the system to assist in performing a task. Two tasks were used: (1) the design and layout of a 4 bit adder and (2) the use of a graphic version of KL-ONE to browse through and modify a KL-ONE network.

The two participants in each task communicated in typed English over a computer link. Visual material was projected on a screen by two overhead projectors so that the "system" could draw pictures and the user could point to them, as shown in the diagram below and detailed in [11].

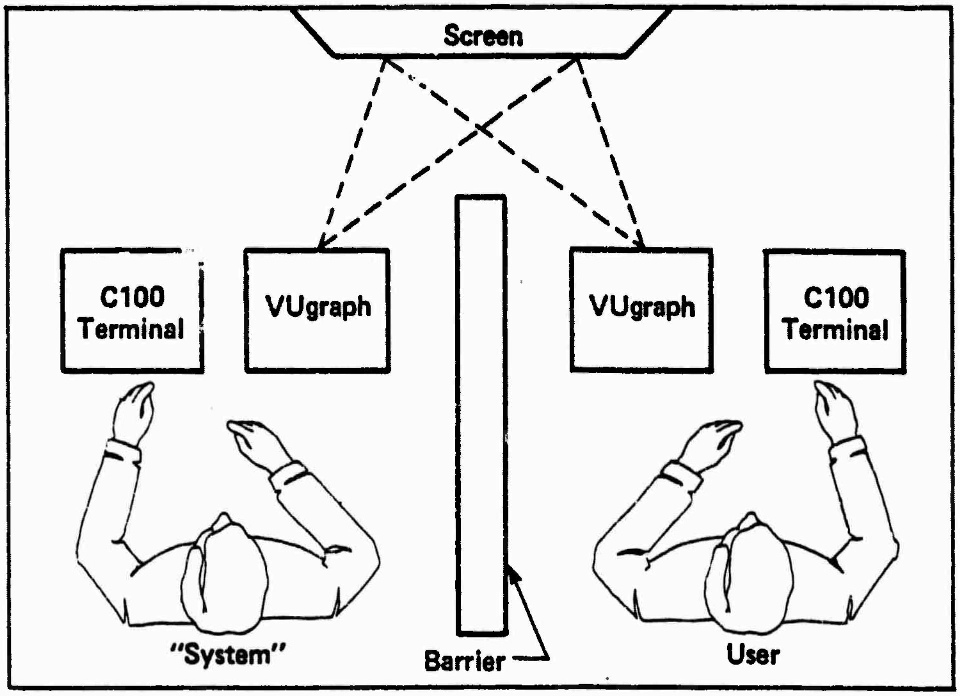


FIG. 1. PROTOCOL SESSION LAYOUT

To maintain as much as possible the user's feeling of communicating with a "system" rather than a person, and to avoid the possibility of communication via body language, eye contact, etc., the two participants were not visible to one another. We collected protocols that consisted of transcripts of six dialogues, each nearly 2 hours in length, as well as all the pictures drawn during each session and all the places in the pictures that users pointed to.

We studied these protocols for typical user behaviors and sought out sequences of behaviors which a system with natural language and graphic output could potentially recognize and respond to. We then devised a prototypical scenario based on the behavior sequences we observed in the protocols. The scenario we present below is prototypical of users' interaction with the KLONE-ED domain. This scenario draws upon the kinds of things that either the user and the "system" said to one another directly, or that represented some higher level phenomenon, such as a clarification exchange. For some behaviors, only a simple capacity was included: while we included some capability for recognizing user errors, the protocols indicate that people can recognize and correct errors far beyond what we could envision for research in the near future. For example, in the protocols, the "system" could easily correct unusual spelling errors as well as misconceptions the user had about the KL-ONE database that was being used. Likewise, we excluded the use of a standing orders as a means of answering questions by the system; in the exchange below, the user's answer to the system question is just such a case:

User: Give that individual [KL-ONE concept] golf as the filler for hobby.
System: Should I create an IROLE for hobby?
User: In general I user [user's spelling!] filler to denote an IROLE.

During the construction of the scenario from our protocols, we amended the scenario twice due to limitations of current tools and theories to handle some of the behavior we wished to model.

For example, we dropped the use of metonymy in the original version of exchange [3] (which was "Ok, now make an individual PERSON with first name "Sam" and last name "Jones."); in exchange [5] we changed the system's response from "Neither is a role. IS THE INFO YOU ARE TRYING TO ADD ABOUT EMPLOYEE BENEFITS?" to the one in the scenario below because the original form assumed that the system could reason about the specific semantics of pension plans, etc., as being parts of employee benefits. Neither capability was possible except with special purpose rules that would not generalize to other related phenomena.

A large part of our effort has been directed at analyzing the prototypical scenario we created in order to design and implement a KLONE-ED system. In particular, by considering not only the scenario exactly as played but also alternative user and system responses at various points, we can view the scenario as a particular path through a large tree of possible dialogues. This means that we can investigate not only what can be said, but what can't be said at a variety of points in such dialogues. This type of consideration illuminates the system capabilities that are necessary to recognize alternative responses and to respond in a intelligent fashion. While not necessarily uncovering all such capabilities, at least the most important ones become clear.

In the analysis for an initial design phase, we determined major modules of the system design. We then stipulated the I/O behavior for each module. This analysis allowed us to clarify our representations, to discover their weaknesses, and to determine what each module must do. Such stipulations gave us an initial design so that we could begin considering how each module would perform its function.

The scenario is half of a feedback loop in a system development cycle that also includes a system design. The design of the system is fed by what we observe in the scenario, and some aspects of the system design, such as the syntactic-semantic-pragmatic cascade, forced us to see features of the scenario we otherwise would have missed.

To refine a system design, we used available research tools to check proposed I/O representations produced by the major system modules. We relied on the RUS/PSI-KLONE interface to check out proposed syntactic and semantic I/O; our current theory of speaker meaning provided a means of determining the speaker's intention which could be used for planning a response; the cascaded control structure of RUS and PSI-KLONE formed the basic system organization for the modules; finally the KL-ONE representation language served as the underlying I/O language for all modules. With these tools we were able to recognize syntactic ambiguities we did not foresee, and to formulate criteria for the operation of the pragmatic modules (recognition of speaker intention, response planning and reference identification). We discuss these in more detail in a later section.

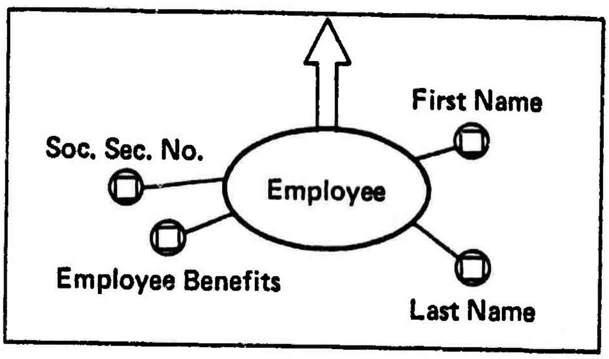
Since the scenario below (which is a composite drawn from 74 pages of actual protocols) reflects our judgements about what we could foresee a near-term system being able to do in the KLONE-ED world, a few words about that domain are in order. Our KLONE-ED system is intended to be both an interactive layout assistant to a user who wants to put a picture of a KL-ONE database on a screen, and a graphics editor that enables the user to construct and modify visual data that will be stored for later recall in the database.

These two uses vary not just in their intent and effect but also in the common sequences of utterances a user would mak to enlist and manage the system's help. Although a sin ar utterance might be made in both types of task interactions, the appropriate system response for each can differ. For example, consider the utterance "Remove this concept." In a layout task, the user may mean that s/he wants the the concept to be suppressed on the screen, but in editing the user may want to have the concept deleted from both the screen and the database. Differentiating between these situations and recognizing when clarification is necessary is of prime importance for the KLONE-ED system.

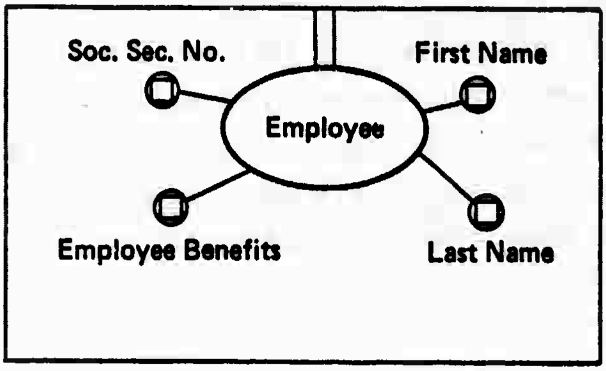
Below is the scenario that we constructed from the protocols for the KLONE-ED domain. We will discuss the first six exchanges in detail in upcoming sections.

THE SCENARIO

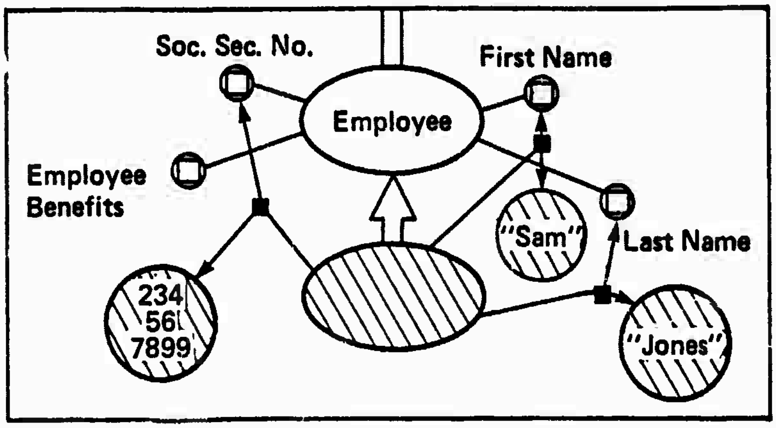
[1a] User: Show me the generic concept called "employee".
[1b] System: OK.



[2a] User: I can't fit a new individual concept below it. Can you move it up?
[2b] System: Yes.



[3a] User: OK, now make an individual employee concept whose first name is "Sam" and whose last name is "Jones". The Social Security number is 234-56-7899.
[3b] System: OK.



[4a] User: Is there a role on employee called "retirement fund" or something like that?
[4b] System: No there isn't. What information are you trying to add?

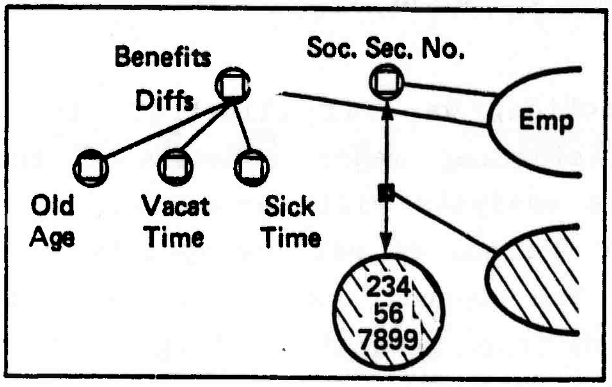
[5a] User: How about a role called "pension program" or "pension plan"?
[5b] System: Neither is a role. Would you like to see all the roles on employee?

[6a] User: No, I'd like to see the structure below employee benefits.
[6b] System: Shall I empty the screen and save the current display?

[7a] User: No, I want the stuff here. (points to upper left

corner of the screen)

[7b] System:

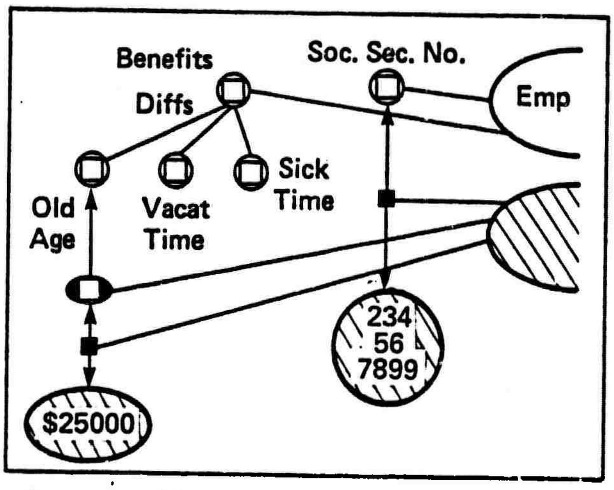


[8a] User: What's the value restriction on "old age benefits"?
[8b] System: A dollar amount.

[9a] User: Give the ic $25000 as the value of "old age benefits" under "employee benefits".
[9b] System: By under employee benefits, do you mean to put the particular roleset and irole or just the irole?

[10a] User: Can you display only the roleset?
[10b] System: No.

[11a] User: Use the irole. No, nevermind. Display the roleset and irole.
[11b] System: Done.



[12] User: Now I need some figures on overhead and R&D.

3 Observations on the Scenario

In this section, we will illustrate the analysis of the scenario by considering major aspects of the first several exchanges. This analysis will demonstrate how our methodology has affected our design, as well as specify a large part of our system design. The appendix of this report presents the KL-ONE representation of the semantic interpretation of the user's utterances in each exchange.

3.1 Exchange 1

Syntax and semantics: In utterance [1a], the user's description of the concept, with quotation marks around the word "employee," demonstrates only one way to deal with special terms.[2] In the actual protocols, users began conversations in this way but quickly abandoned the use of quotation marks (without apparent difficulty for their human conversational partners). We have retained quotation marks throughout the composite scenario; their use simplifies the operation of our parser, since it is possible to process the quoted material as an arbitrary string rather than detecting, defining, and appropriately parsing new, undelimited names.

All noun phrases in the scenario are semantically marked

[2] In figure 1 of the Appendix, we use the full description of this concept, namely a KL-ONE concept with a role of label whose value is a string with a role of spelling with a value "employee." In later figures wherever possible we abbreviate this as the KL-ONE concept with label "employee."

with a special discourse marker that indicates that they represent some entity that is described in the discourse (not necessarily corresponding to an entity in the database or on the screen). In the figures of the Appendix the concept representing this information is marked as "D(escribed) in D(iscourse)." These discourse entities are later used by the reference mechanism to determine whether there is a referential relationship between the original noun phrase and some object in the real world. These ideas are more fully described in [13].

Discourse capacities: An analysis of alternative previous contexts has convinced us that the user's first utterance, [1a], is not a direct imperative in the "command language for computer system" sense of the word. "Show" does not detail just how the system is expected to act (it could draw a picture or print text, for example); the user expects the system to judge what meets the "show intention" because the user and the system share mutual knowledge (see references [1], [6] and [4]) of the system's overall capabilities, at least in terms of their effects. Two alternative forms for this utterance which could just as easily have been said are:

- I want you to {show, display} the generic concept called "employee."
- I want to see the generic concept called "employee."

The second alternative utterance illustrates that the user's request can be stated even more completely in terms of effects desired rather than acts to perform.

Our analysis [10], [12] shows that the system must consider four sources of information before choosing a response: the speaker's intention, the speaker's overall goals that are gathered by the system during the entire discourse, clarification from the user about what to do, and the system's own "non-dangerous" defaults. For example, because the user does not say exactly where to show the concept, the system is expected to judge for itself or to ask for clarification, but the former is far more desirable. Choosing a mid-screen position by default is reasonable because the user can undo this action; asking the user for clarification interrupts the discourse and seems to occur only when all of the default choices have undoable side effects.

Awareness of the speaker's overall goals helps to ensure proper placement of concepts on the screen. Sometimes, as in this dialogue, the system cannot know the user's goals; it does not know that the user is editing the database and needs room to add new screen concepts. The user has not stated this "high level" goal (which is one way the user might have begun this dialogue), and the system cannot deduce it on the basis of the one utterance, because "showing" is evidence for many possible courses of action by the user (others include browsing the network or layout of a part of the network on the screen). Again, the system might have interrupted to ask the user for an explicit statement of the goals; however, intelligent human assistants interrupt infrequently, even though failing to ask leads to the very common kind of error correction in utterance [2a].

<u>System Organization</u>: The processing for syntax and semantics that we envision makes use of the PSI-KLONE interface [3] for cascading syntactic and semantic decisions. The syntactic component is able to call upon semantic information to guide the parsing process. We also envision the need for extending the cascade to include pragmatic decisions so that semantically

ambiguous sentence meanings can be resolved, rather than passing several alternate interpretations of a sentence to pragmatics.

The definite noun phrase in [1a] indicates that the item referred to ("the generic concept called 'employee'") is mutually known to user and system. The system must consider two sources for the referent of a definite noun phrase: the screen and the database. Pragmatic information will determine that in this case the database is the source of reference since the screen is empty. In other cases, the interpretation of the phrase depends not only on the contents of the two sources but also on what the system is to do with the referent. For instance, if the user requests that something be moved to the right of the generic concept for "employee" when the screen has one representation of the concept and the database another, the interpretation depends upon which concept can be described directionally. Such cases, as well as later exchanges in the scenario, have convinced us that reference interpretation must be delayed as long as possible (sometimes even until a response is enacted) to take advantage of as much pragmatic information as possible.

3.2 Exchange 2

Semantics: An ambiguity of scope occurs in the first sentence of [2a]. The user is either talking about some one particular individual concept, or is saying that none of the individual concepts will fit below the generic. Either way, the scope ambiguity matters only if the system has some means to distinguish these two kinds of individual concepts. If, for example, the fitting action differs depending on the individual concept being fit, then the system must choose the more plausible of the two interpretations. Suppose that the system knows of only one "fit" action; then the scope ambiguity here can be ignored until a response is required with the second sentence of the utterance. When the system must respond to the question of moving the generic concept up, the distance it moves it depends on the size of the new concept. At that point, the system must take into account the scope ambiguities only if it can vary its moving by the kind of individual concept being moved. If it cannot, the scope ambiguity is best ignored.

These observations about the system's capacities lead us to conclude that determining the <u>scope</u> of an ambiguity should be delayed until there is some choice that depends upon it. Then, if no choice ever does, the scope can remain unspecified. This is stronger than saying that <u>disambiguation</u> should be delayed until some choice depends on it; we are actually delaying the explicit identification of the alternatives. This suggests that the semantic interpretation of a sentence may leave scope undefined as long as there is a communication path back to the semantic component to get more information about the scope should some choice arise for which additional semantic information could give decisive evidence.

<u>Discourse capacities</u>: With utterance [2a], "I can't fit a new individual concept below it. Can you move it up?", the user indicates to the system that there is a problem with the display. This exchange is typical of human dialogue: something goes wrong, so one of the conversants describes the error and what to do about it. It is important that the user believes his partner is intelligent--otherwise there is no point telling the partner what has gone wrong; instead the user would simply command the assistant to move the generic concept up.

To respond to the actual utterance, the system must make use of rich knowledge about the user's behavior, and about the user's intentions for the system to use that knowledge. The sentence "I can't fit a new individual concept below it" not only states the problem but also states indirectly that the user had planned to put a new individual concept below the generic one; both propositions must be recovered by the system, and it must also recognize that the user is going about correcting the error.

A part of the system's response depends upon whether it can distinguish some action to take based on the difference in semantic scope of quantifiers discussed above. We plan to experiment with a version that can distinguish the two scopes and with one that cannot. The latter is simpler and will be explored first; the former requires additional research into the nature of a cascade of information between semantics and pragmatics.

The second sentence of the utterance, "Can you move it up?" provides a request about what the system should now do to correct the situation. The request is stated as a question, but not simply because it is polite to do so, or because one strategy for requesting an action is to ask about its precondition. With the question the user is simultaneously cutting off some of the system's options (such as erasing the screen to make room), and leaving open the possibility that the system has some better option that the user hasn't thought of (such as using a window for the new concept). For the system to judge what constitutes a better option, it must take into account the user's goals and some judgements about esthetics of the screen. We have no idea how to quantify the latter at this time, but we have designed a model to deduce the user's goals.

The two uses of pronominal reference offer no particularly difficult problems. By following our notion of delaying reference interpretation as long as possible, the system would have evidence for just what kind of thing the "it" must refer to. In the Appendix "it" is interpreted as both a pronoun and an inanimate object. Because of the domain, "it" will always be an inanimate object and can be so specified by the semantics; but in general by using a delayed reference technique, our system can infer the type of object referred to by "it" from the pragmatic context of the sentence.

System organization: To delay the determination of scope of ambiguity, we must provide a system design that permits communication feedback to the semantics from the pragmatics with a language in which to ask questions about scope. Hence the cascade between syntax, semantics and pragmatics must allow for feedback at times other than during the initial process of interpreting the phrases of the utterance.

The second sentence of [2a], "Can you move it up?", provides a means of fixing the problem stated in the first sentence, "I can't fit...". Such multi-sentence utterances are common. They illustrate the need for a system cascade that is capable of passing through larger units of discourse than single sentences. For this utterance in particular, the system must notice the second before it plans a response to the first one. If this were not the case, the system might respond to the first sentence by volunteering to move the concept or erase the screen. Once the second sentence is uttered, such a response is inappropriate. Our system will allow multi-sentence utterances by delaying its planning of a response until it has processed all the parts of the user's utterance.

Domain Specific Knowledge: The entire exchange in [2] indicates that the system must have detailed information about directions (such as "below"), regions, and locations. Much of this information is used implicitly in moving objects around on the screen, and in understanding what region is defined by phrases such as "below it." In the KLONE-ED domain, words such as "below" have meaning in terms of relationships such as SuperC, and result in completely different interpretations depending on the objects involved.

3.3 Exchange 3

Semantics: Verbs such as "Make" require a quotation capability in semantic interpretation. The semantic object of the verb, namely, "an individual concept whose..." does not name an existing object but rather is a description of the kind of object that will result from the making action. Most verbs (e.g. give) are ambiguous about whether their objects refer to existing entities or can be interpreted as descriptions[3]. Since some verbs are unambiguous in this regard (e.g. make, delete), it is useful to make this information available to later processing. To indicate the semantic distinction, we have introduced a quotation notation to semantics for object positions of some verbs.

The KL-ONE description of the semantics for [3a] that is

---

[3]Barwise and Perry [2] use the terms "value laden" and "value free" to express the distinction we are drawing here about noun phrases being used either as descriptions or references to objects.

given in the Appendix presents only one of the two possible readings for this sentence. The verb "make" is ambiguous between a sense of making screen objects (the one given in the Appendix) and making database objects. While a complete cascade would possibly allow the two senses to be disambiguated, the details of how such a communication would proceed are not yet clear. This utterance is one case in which several alternative semantic interpretations would need to be passed through to the discourse interpreter.

To interpret in a general way the relative clause "Whose first name is 'Sam' and whose last name is 'Jones'" requires an overall solution to a general class of semantic forms that remains beyond our grasp. For our domain, we are able to rely on the particular definitions of generic and individual concepts, available from a syntactic and semantic taxonomy expressed in KL-ONE, to produce a proper semantic interpretation. That is, we use the domain-dependent fact that the locution "the <concept> whose <x> is <y>" can mean the concept that has a role named x with role value y. Similarly the semantic interpretation for "The Social Security Number" can be constructed using the fact that "social security number" is the role name of some concept in the network.

Discourse capacities: The structure of the discourse and the speaker's plan is reflected in the use of "Ok, now." Reichman [7] observes that such clue words indicate the shift between boundaries of the discourse structure. In particular, here the shift is to indicate the completion of the error correction discussion and the resumption of the editing task. Our model meshes the discourse structure elements with the speaker's plan. The plan defines and limits the set of context spaces that the hearer uses to model the speaker's discourse, and it determines where the conversation returns to. Thus when the speaker signals to the hearer that a boundary has been reached (by "Ok, now") and the conversation is shifting, either the discussion moves to the next step of the plan, or, in the case of interruptions, to the last step of the plan before interruption.

That the plans and discourse structure are interconnected this way is also evident from the implicit request in utterance [3a] for where to put the individual concept. That the making is intended to mean "Put it on the screen," is not evident from the sentence alone, but rather from the previously recognized overall goal of editing and placing a new concept below the generic employee concept.

The sentence "The Social Security Number is 123-45-6789" contains an ellipsis to the individual concept referred to in the first sentence of [3a]. That there is an ellipsed term is semantically determined by the use of "Social Security Number." The use of a definite rather than indefinite description indicates the user's belief that the item is uniquely specified in the context for both the user and the system. By locating the ellipsed term, and by using the focus machinery and definite noun phrase rules of Sidner [8], the system can determine that the referent of the definite noun phrase is the social security number role of the (as yet non-existent) concept mentioned in the first sentence of [3a].

<u>System organization</u>:A behavior similar to the multi-utterance behavior in [2a] illustrates another aspect of multi-sentence utterances. Here the second sentence serves as an addendum to the first. The strategy of the system delaying its

response until the utterance is completed permits the system to treat the two sentences as a single request for a single, though complex, action.

3.4 Exchange 4

Syntax and semantics: Utterance [4a], "Is there a role on employee called 'retirement fund' or something like that?", is a simple question syntactically except that "employee" is being used as if it were a proper noun. Syntactically, it can be considered an ungrammatical noun phrase that lacks a determiner (or ought to be plural), or it can signal an ellipsis from "the concept called 'employee'". Due to domain information, our parser will treat it as the latter.

The phrase "something like that" remains problematical. We included it in our scenario because it occurred more than once in our protocols and offered us a challenge. The phrase is several ways ambiguous since it could mean a phrase with a spelling like "retirement fund" or a phrase that is semantically similar in purpose to "retirement fund" or even a role related to employee in some other way than "on employee." One method we have chosen to deal with it is discussed under domain specific knowledge below.

Discourse capacities: Although the literal question that is being asked can be answered with a simple "no", the system should use its knowledge about the user's current task in order to be helpful. Since the user has just added three pieces of data, all of which are fillers of roles on a particular concept, and is now asking about another role on the same concept, and has given no indication of a new goal or subgoal, the system should conclude that the user is probably still trying to add information to that concept. In asking the user to make his/her goal explicit, utterance [4b] can ellipse the particular concept in focus.

The indefinite noun phrase in [4a] is not ambiguous between one particular role the user has in mind, and any such role described by the noun phrase (in contrast to "a new individual concept" in [2a]). The noun phrase has only the interpretation that there is some one role described by the phrase "a role on employee called 'retirement fund'," and the system is to carry out the user's intention using that description. An important aspect of discourse interpretation is determining which of several ways a noun phrase is used. The dialogue gives evidence that a general theory of referential versus interpretive noun phrases is needed since both types freely occur.

Domain specific knowledge: In order to interpret "something like that" the system must know what it means for one role name to be similar to another. In this case, using a spelling-correction metric to determine whether or not any of the existing roles on the concept have names like "retirement fund" would be one way of interpreting the user's request. Realistically, the user probably intended the system to understand something about the semantics of the phrase "retirement fund" rather than to treat it as simply a quoted string. In this case, the system would need to be able to find concepts semantically related to retirement fund and to map those concepts back into strings that can be compared with the role names on the original concept (e.g. "retirement trust", "old age benefits", "retirement plan", "pension fund").

3.5 Exchange 5

Here the user, by ignoring the system's question, declines to make his/her plan explicit without actually saying so. Between equal human speakers, ignoring a direct question violates some social conventions, but since the machine is not the user's equal, the user can rightly ignore the question. Instead the user continues to ask for information via utterance [5a], "How about a role called 'pension program' or 'pension plan'?".

Syntax and semantics: The question form "How about..." is quite informal and is not often encountered in written form. Because it puts a heavy burden of interpretation on the listener, a human would not be likely to use it in a conversation with a computer system unless the system had done a great deal to encourage short, natural communication. Note that other ways of phrasing the same question ("Is there a role called...", "Does it have a role called...", etc.) are much longer. A "how about" question is much more general than a "who", "what", or "how" question. It demands interpretation in context, and requires the system to map the current question onto the interpretation of the previous question (or sentence), finding the appropriate substitution.

Discourse capacities: Although a simple negative answer would suffice here, a system that is trying to be helpful should suggest a course of action that will attain the user's goal when it is clear that the user is having trouble reaching that goal. Such helpfulness is not merely cooperative--the user expects it.

In the last two utterances the user has questioned four different possible role names for the employee concept, although none of them are actually attached to the concept. The system has enough evidence at this point to conclude that the user is searching for a particular role on the concept (as a subgoal of the goal of adding a filler to that role). The user intends for the system to draw such a conclusion, and, as is presumed throughout the scenario, to help the user in whatever way it can. Hence the user intends for the system to use its knowledge of the user's goals to determine how to be helpful. Since not all of the role names of the concept are being displayed at the time of the user's question, and since the user can find a particular role by looking at the set of roles, it is reasonable for the system to generate the suggestion in [5b], "Would you like to see all the roles on employee?"

Just what the system must provide is limited to what it can reasonably conclude about the user's goals. Thus the system can conclude that there is some role the user is trying to find without being able to deduce just which one. Its response is then based on this limited conclusion. A system with more sophisticated reasoning ability about employees might recognize that pension funds, pension plans, etc. are a particular kind of employee benefit, and since the user has mentioned so many of these semantically similar names, he/she could intend a sophisticated system to conclude that the user wants the kind of employee benefit loosely described as a "pension fund" or the like. We judged that a reasoner general enough for this conclusion was not available and could be considered as a future problem. Meanwhile, since the current system is not able to recognize such a similarity, the user cannot realisticallyt intend for it to provide the proper description; instead the user can intend that the system respond using its more limited reasoning capacities.

3.6 Exchange 6

Syntax and Semantics: [6a] "No, I'd like to see the structure below employee benefits," is two ways ambiguous both syntactically and semantically. The syntactic ambiguity concerns the attachment of "below 'employee benefits'" to either the sentence object or verb phrase. The semantic ambiguity concerns the interpretation of "the structure below employee benefits" as a database or screen structure. The syntactic-semantic-pragmatic cascade can eliminate such ambiguities. The approach of delaying the unwinding of ambiguity is not feasible here because the system must act on the request and hence must know the sentence structure and content before it proceeds.

Discourse Capacities: Like exchange [5], at the start of [6a], the system has an expectation that the user will answer its questions, but this time the expectation is fulfilled, with a simple"no." In terms of the structure of the discourse, the answer ends the question exchange and returns the discussion to the level of the user's two questions. However, the return is not a case of the popping phenomenon explored by Grosz [5] because there are no incomplete goals remaining to be popped back to. The system must be aware that the discourse structure contains a return because the return will influence how it resolves the syntactic and semantic ambiguities recognized by the parser and semantic interpreter.

The basic intention in "I'd like to see..." is for the system to use its relevant capacities to enable the seeing. It must also recognize the proper interpretation of the object of "see" and produce the correct database object to draw on the screen. In attempting to plan to draw, the system will recognize a space problem and plan to ask for help on how to proceed. It can ask about two alternatives: empty the screen and display the new structure, or to add to the existing screen. Rather than state both choices, it aims for conciseness and chooses the one it believes is unknown to the user.

<u>System Organization</u>: The cascade architecture of syntax, semantics and pragmatics can decide that two of the three possible interpretations of [6a] are meaningless: one is impossible because "the structure" would refer to employee concept and be a request to put it below its own role, and a second is odd because the screen structure below employee benefits (the ic 123-45-6789) is already visible on an uncluttered screen. By eliminating these two readings early in the interpretation process, the system saves itself having to juggle deductions about all three possible interpretations and then making judgements about the likelihood of each.

4 Conclusions

Below we present our conclusions about the methodology used in this case study and some of the necessary linguistic and non-linguistic capacities for the KLONE-ED system based on analysis of the full scenario.

<u>Syntactic and semantic capacities</u>:

The system must be able to use syntactic, semantic, and pragmatic knowledge to judge ambiguity and to choose (almost always) a single interpretation for an input sentence. This representation may, however, embody some remaining ambiguities.

Both the syntactic and semantic components must be rich enough to represent and interpret noun phrases that serve as quoted descriptions of an object (ideally, with or without explicit marking in the input). For example, it must be possible for the user to refer to objects that do not exist yet but which might exist at some time in the future, as in "Create an X".

Activities such as pointing to a particular place on the screen while saying a word such as "it" or "there" must be recorded as part of the user's input; the graphic domain, together with the KLONE-ED domain, specifies information useful in assigning a referent to these deictic pronouns.

System Organization:

The syntactic, semantic and pragmatic processors could be cascaded, so that, for example, the syntactic component can call upon semantic information to guide the parsing process while, at the same time, the pragmatic component can begin to work on constituents and provide feedback to the semantics and syntax about possible interpretations.

When the scope of a quantifier is ambiguous, the interpretation of the scope should be delayed as long as possible, perhaps until a subsequent sentence, until a choice must be made which depends on it. With the delay, the maximum amount of information necessary for determining the proper scope can be accumulated.

When various types of referential expressions are encountered, including pronouns and ellipsis, the referent finding process should be delayed as long as possible. Other processes, such as semantics and intention recognition, will collect and represent certain features which the ultimate referent must have.

Multi-sentence utterances require a system organization that marks utterance units in addition to sentence units and delays planning and response until the whole utterance has been processed.

Discourse capacities:

Pragmatic knowledge of the kinds of user actions, goals, and capabilities must underlie the system's reasoning about both the interpretation of the user's input and the content of the system's reaction. The system must understand its own capabilities and those of the user, and must have a model of the user's beliefs about the system and the current situation. Without mutual belief, intelligent conversation is impossible.

The system must know, and therefore have a representation of, what information is in its internal data base, what information is on the screen (and apparent to the user), and what information the user knows about the database(as a result of previous viewing or previous knowledge of the content of the database).

The system must be able to recognize and reason about the user's high level goals from the evidence explicitly or implicitly available in the dialogue. Complete mutual knowledge and belief cannot be assumed.

The system must recognize what the user wants the system to do based on the user's utterances, mutual beliefs about the system's capacities and default behavior, the context of the previous discourse, and the user's overall goals.

As the system becomes more helpful and intelligent, it requires more capabilities and hence has more choices for acting. It must then choose from many possible alternative responses, all the while bearing in mind the user's intended responses.

Certain interjections reflect changes in the discourse structure. The system must recognize these interjections and their import, and must use them itself when necessary.

Domain Specific Knowledge:

It is necessary to have a fairly complete, possibly informal, categorization and description of the kinds of knowledge the user and the system have about their mutual capacities and tasks. For example, the system has to know that the user can edit both screen objects and data base objects, that the user has a hierarchy of goals that the system will have to infer, that the user makes and corrects errors, and that the user will adhere to conversational conventions. The user, on the other hand, must know at least a subset of the system's capabilities, including the system's knowledge of the user.

A goodness metric is needed for judging the placement of screen objects and such notions as "cluttered", or "balanced." This is needed not only for aesthetic reasons but also in order to understand what is intended by an utterance like "show me X" where X is already somewhere on the screen.

The system must also have knowledge of locations, regions and directions vis a vis the screen and related user acts. It must be able to determine, for example, when the user points to a particular place on the screen, whether s/he is indicating a location, a region, a screen object, a constellation of screen objects, a data base object, a relation, or some other kind of entity.

Methodology:

Although the construction of a prototypical scenario cannot be entirely free of the problems associated with creation of a sample scenario by the method of introspection, it greatly reduces those problems and results in a compact, easily-understood benchmark against which future progress can be measured.

Preservation of the original protocols is a necessity, as some issues that arise in the analysis of the scenario (particularly those related to alternative actions on the part of either the system or the user) can be better understood by returning to the original data. Similarly, hypotheses about how to handle issues in the scenario can be tested against the protocols before they become a hardened part of the system design or implementation.

APPENDIX A
SEMANTIC INTERPRETATION OF UTTERANCES [1A]-[6A] USING KL-ONE

The diagrams on the following pages illustrate KL-ØNE structures that would result from the semantic interpretation of the utterances produced by the user in the scenario beginning on page 9.

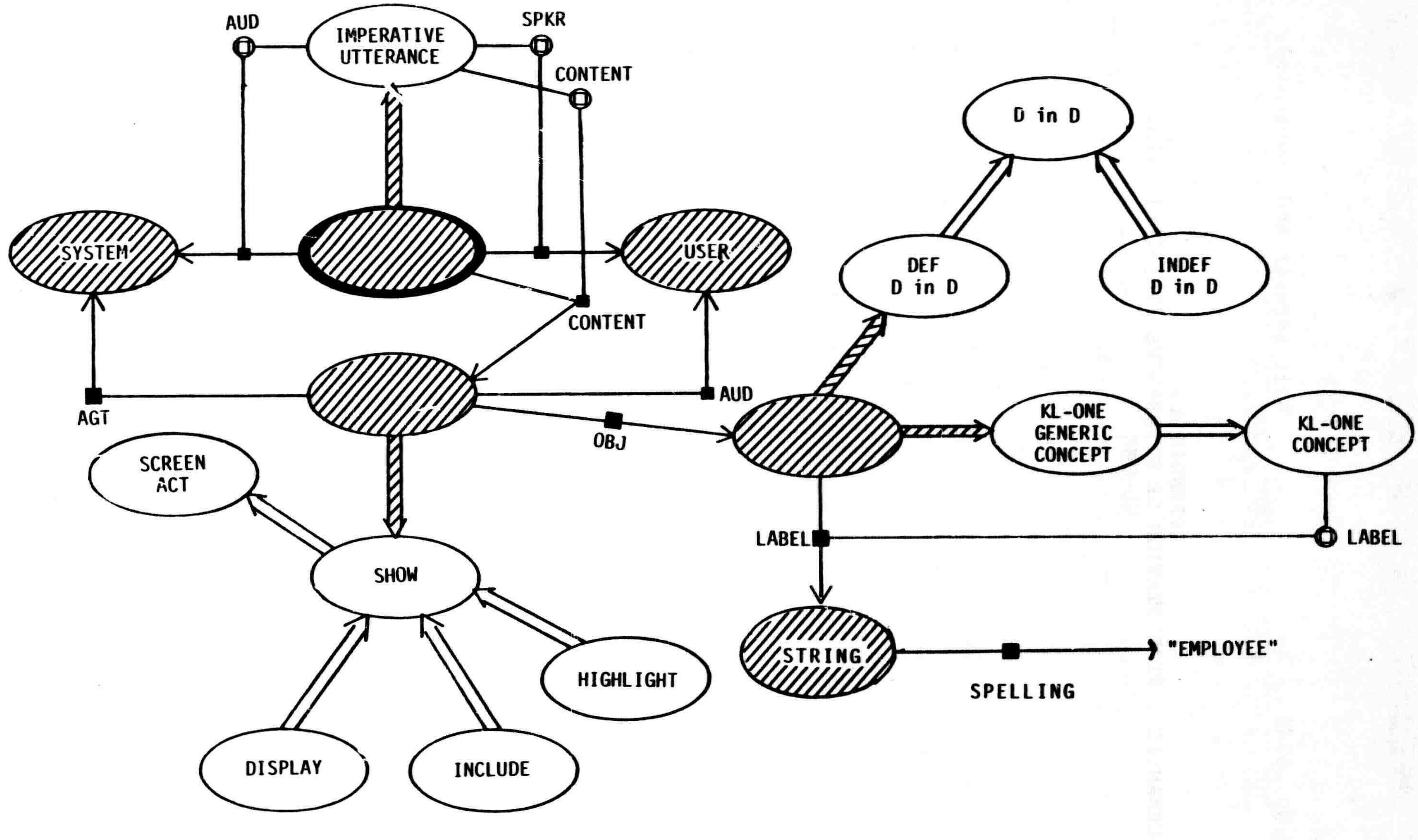


FIGURE 1. SEMANTICS FOR SHOW ME THE GENERIC CONCEPT FOR 'EMPLOYEE'
SHOW ME THE GENERIC CONCEPT WHOSE NAME IS 'EMPLOYEE'
SHOW ME THE GENERIC CONCEPT CALLED 'EMPLOYEE'

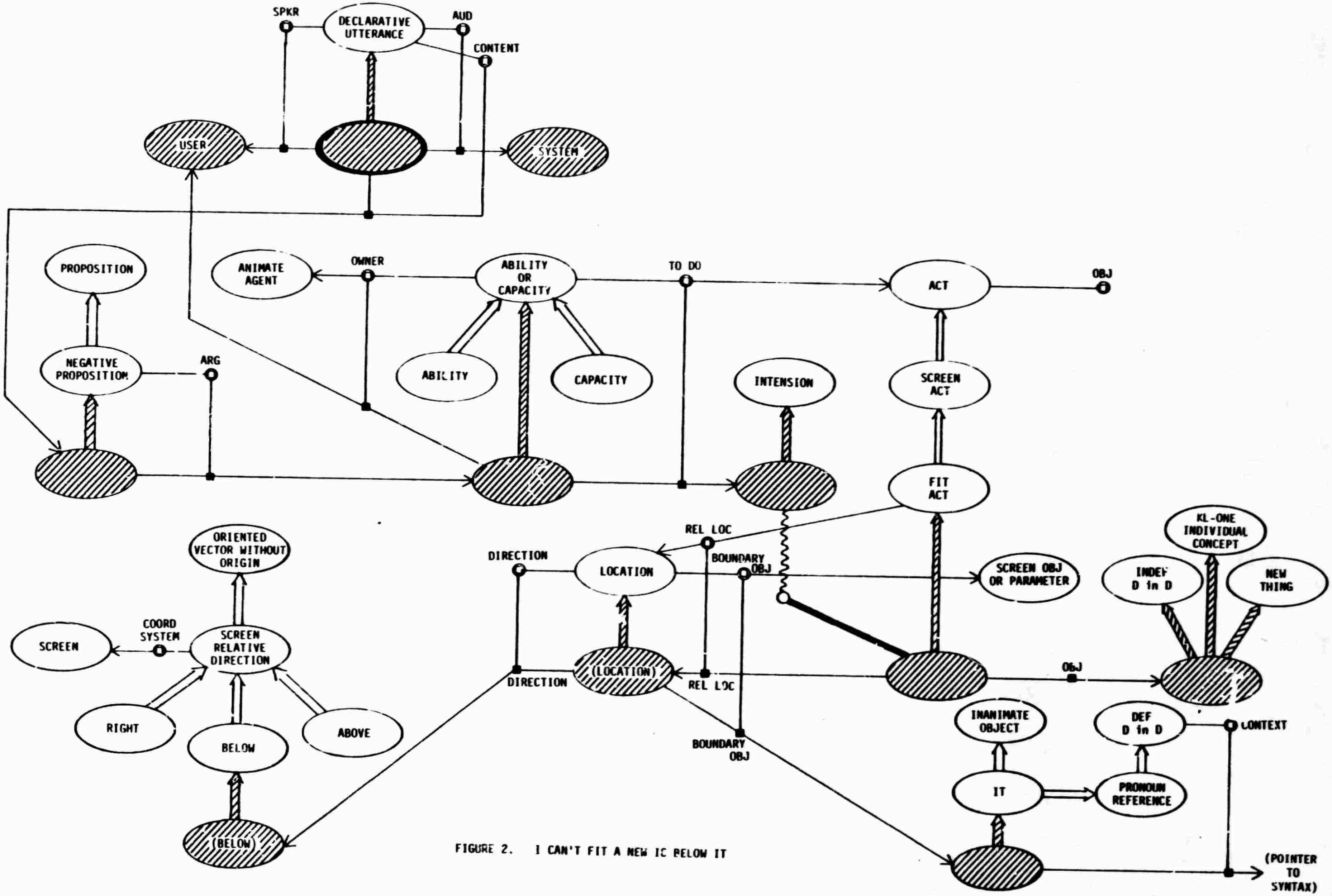


FIGURE 2. I CAN'T FIT A NEW IC BELOW IT

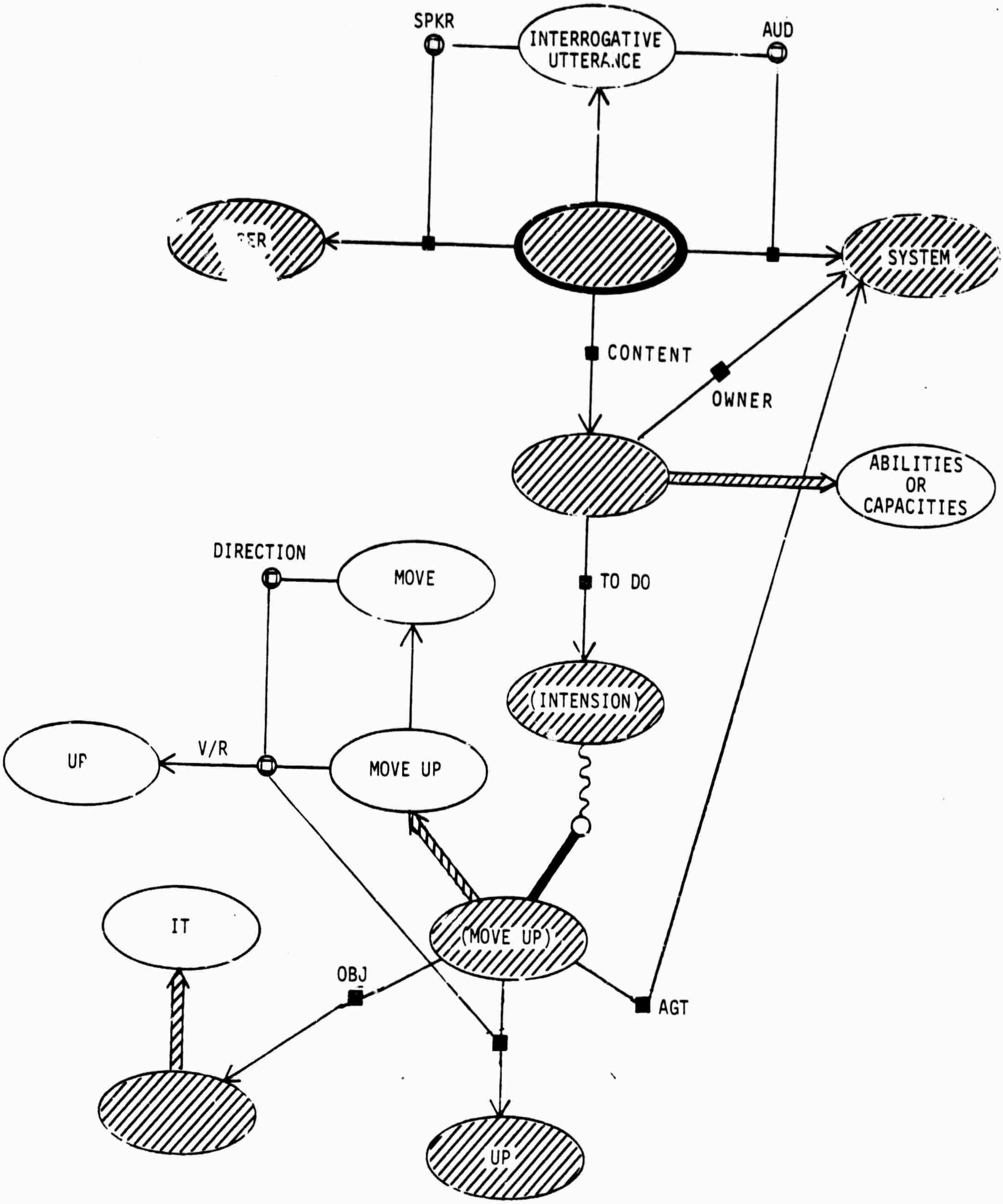


FIGURE 3. CAN YOU MOVE IT UP?

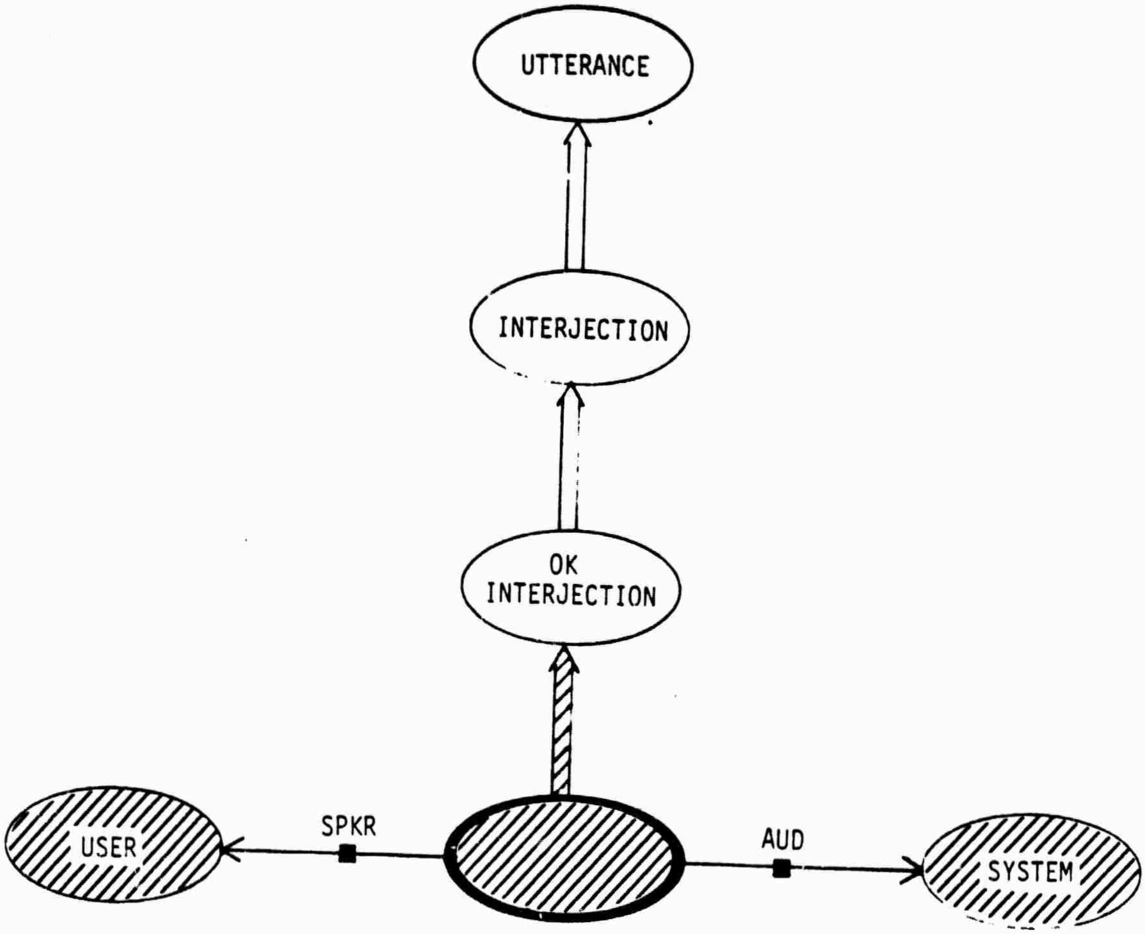

UTTERANCE
INTERJECTION
OK
INTERJECTION
USER
SPKR
AUD
SYSTEM

FIGURE 4. OK

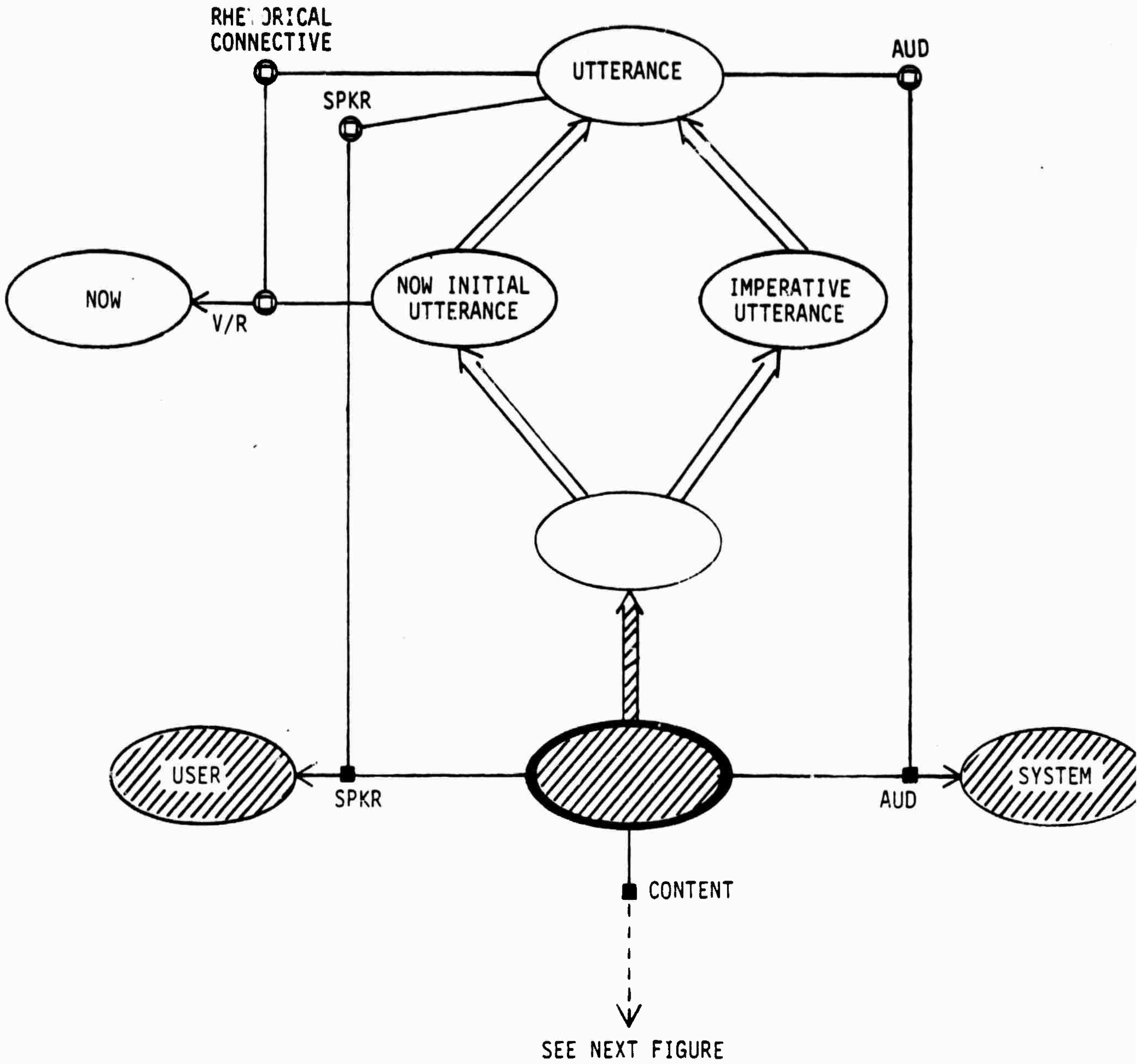


FIGURE 5.

NOW MAKE AN INDIVIDUAL EMPLOYEE CONCEPT WHOSE FIRST NAME IS "SAM" AND LAST NAME IS "JONE

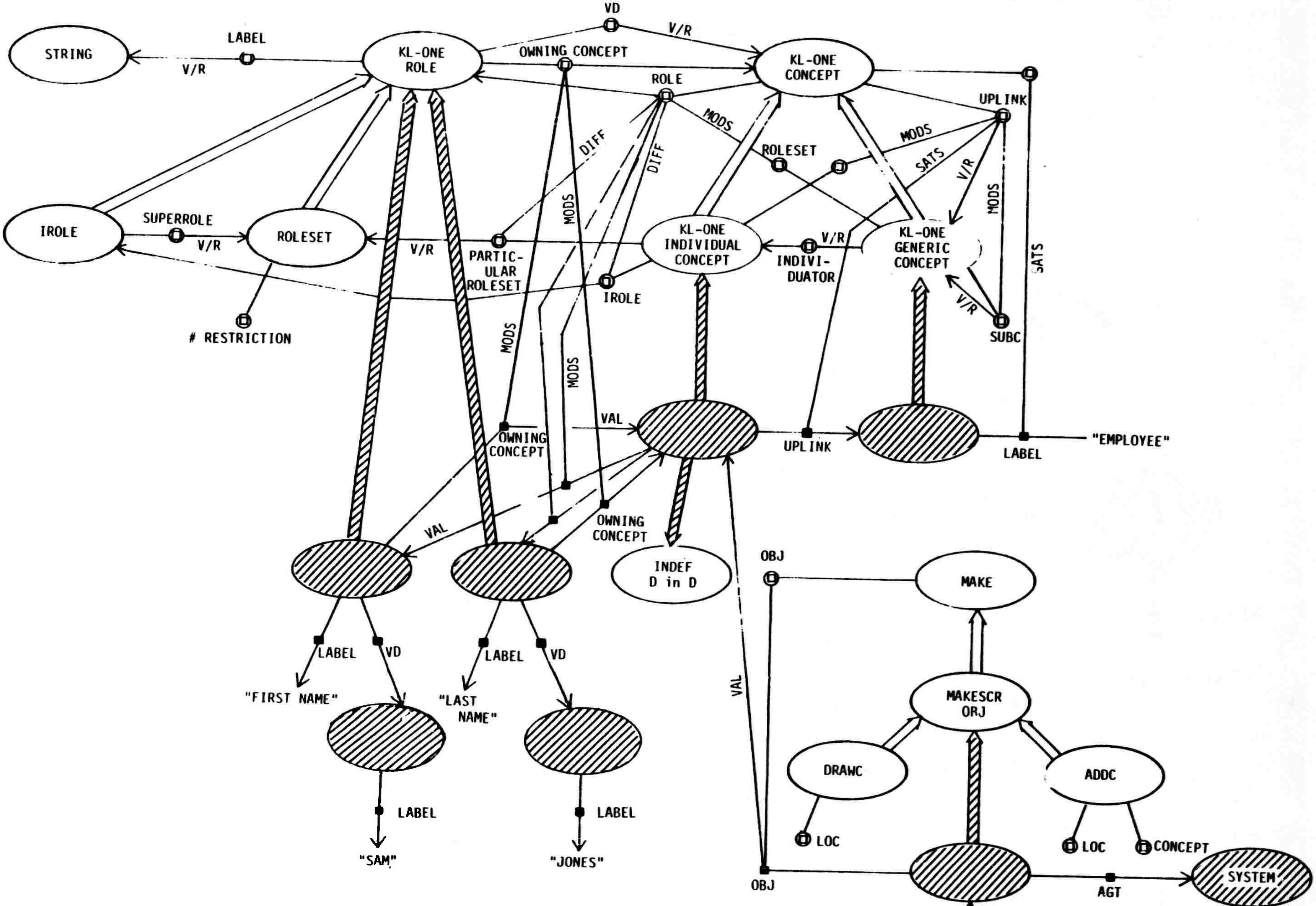


FIGURE 6. (Cont. of) NOW MAKE AN INDIVIDUAL EMPLOYEE CONCEPT
WHOSE FIRST NAME IS "SAM" AND WHOSE LAST NAME IS "JONES"

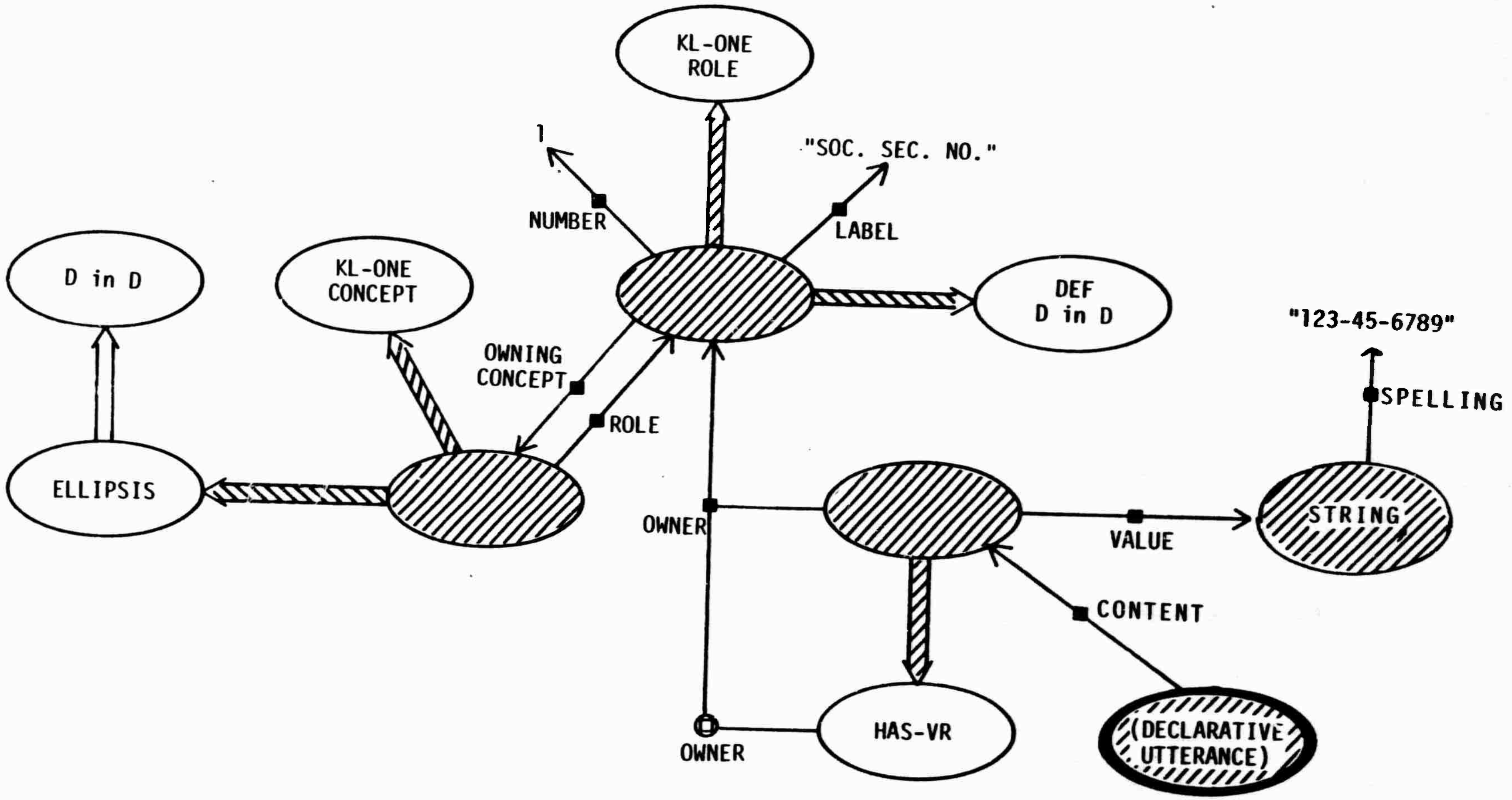


FIGURE 7. THE SOC. SEC. NO. IS 123-45-6789.

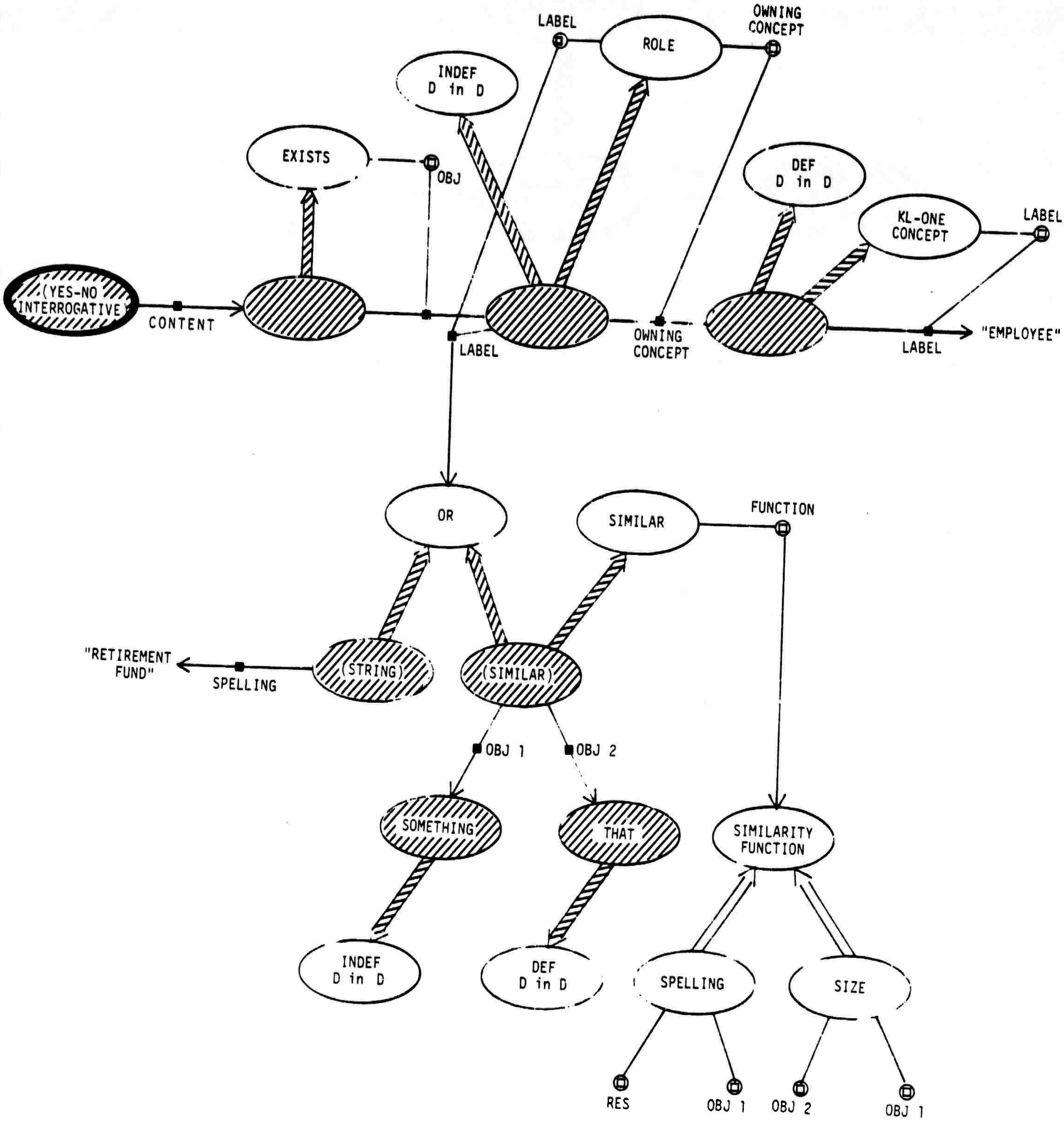


FIGURE 8. IS THERE A ROLE ON EMPLOYEE CALLED "RETIREMENT FUND" OR SOMETHING LIKE THAT?

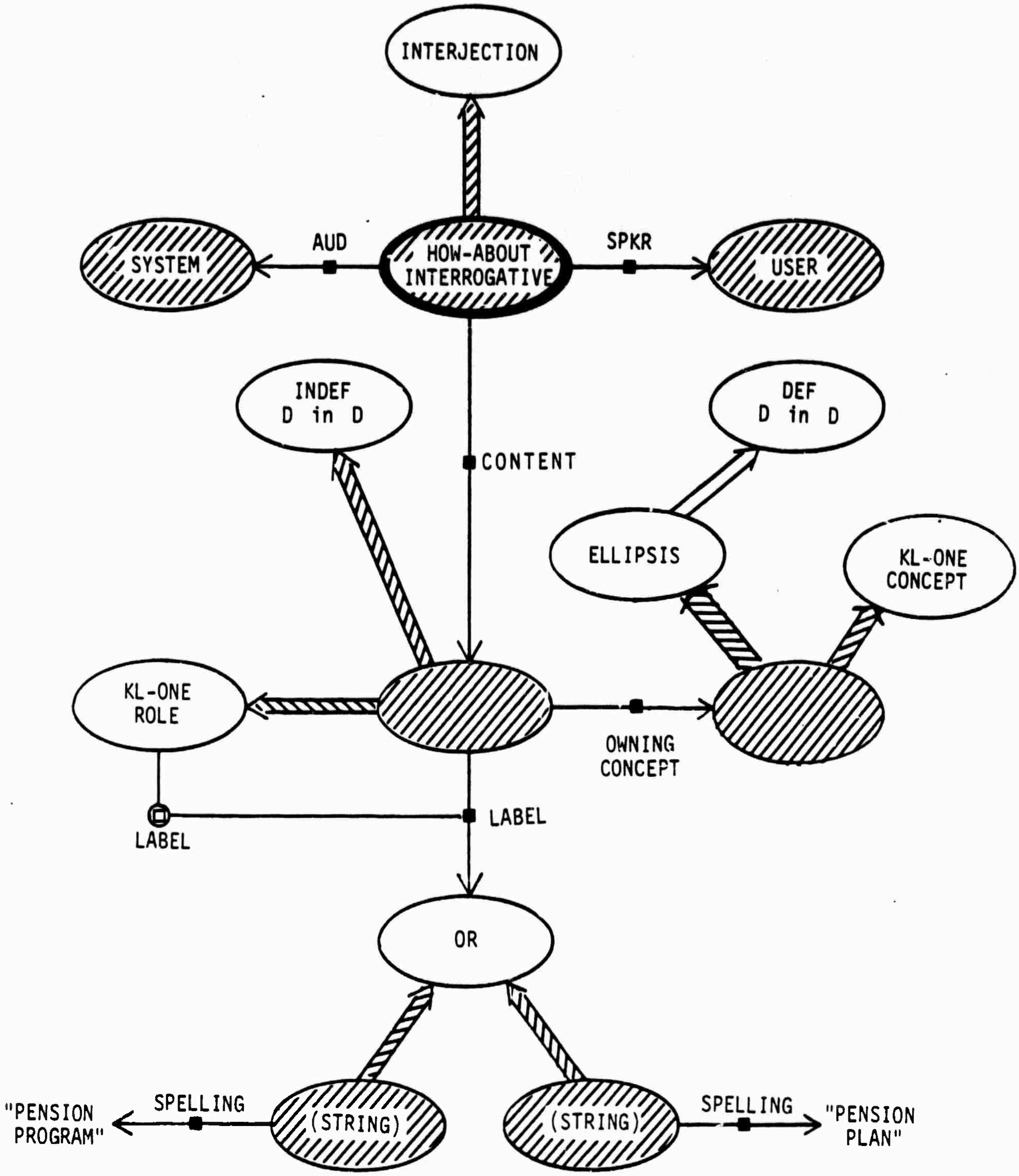


FIGURE 9.

HOW ABOUT A ROLE CALLED "PENSION PROGRAM" OR "PENSION PLAN"?

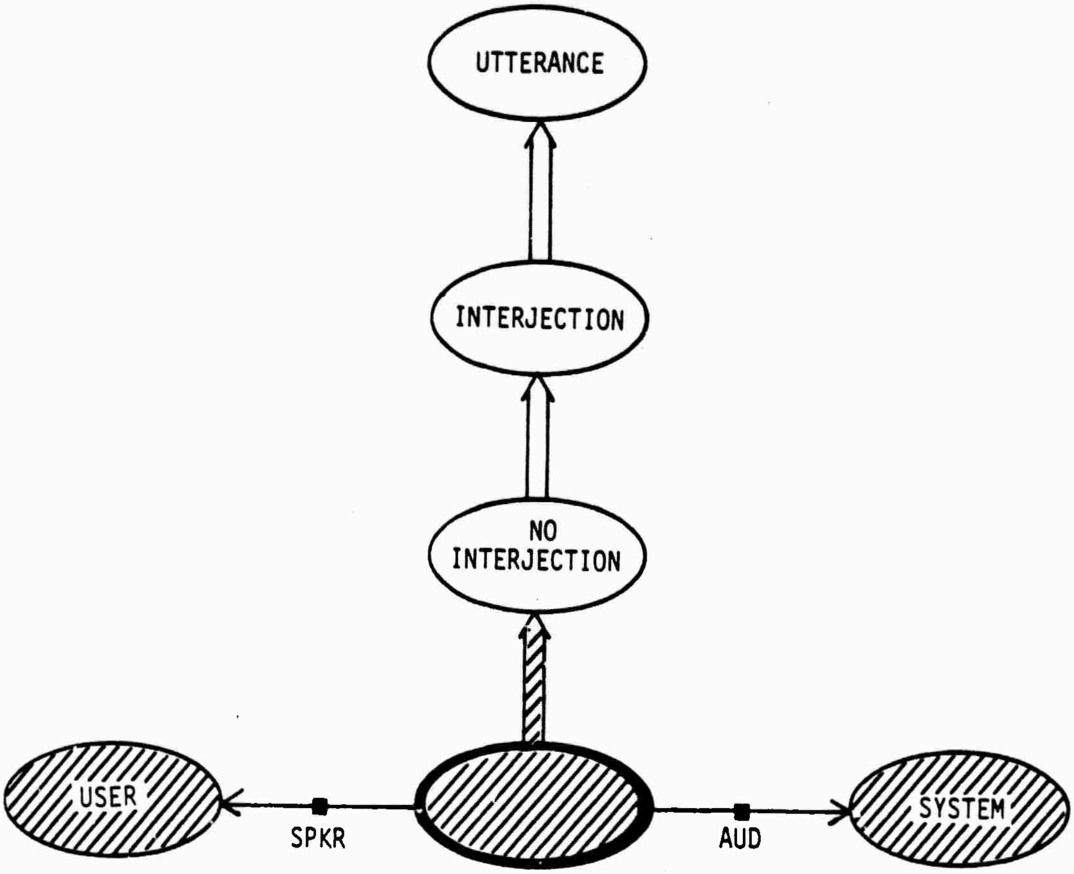

UTTERANCE
INTERJECTION
NO
INTERJECTION
USER
SPKR
AUD
SYSTEM

FIGURE 10. NO

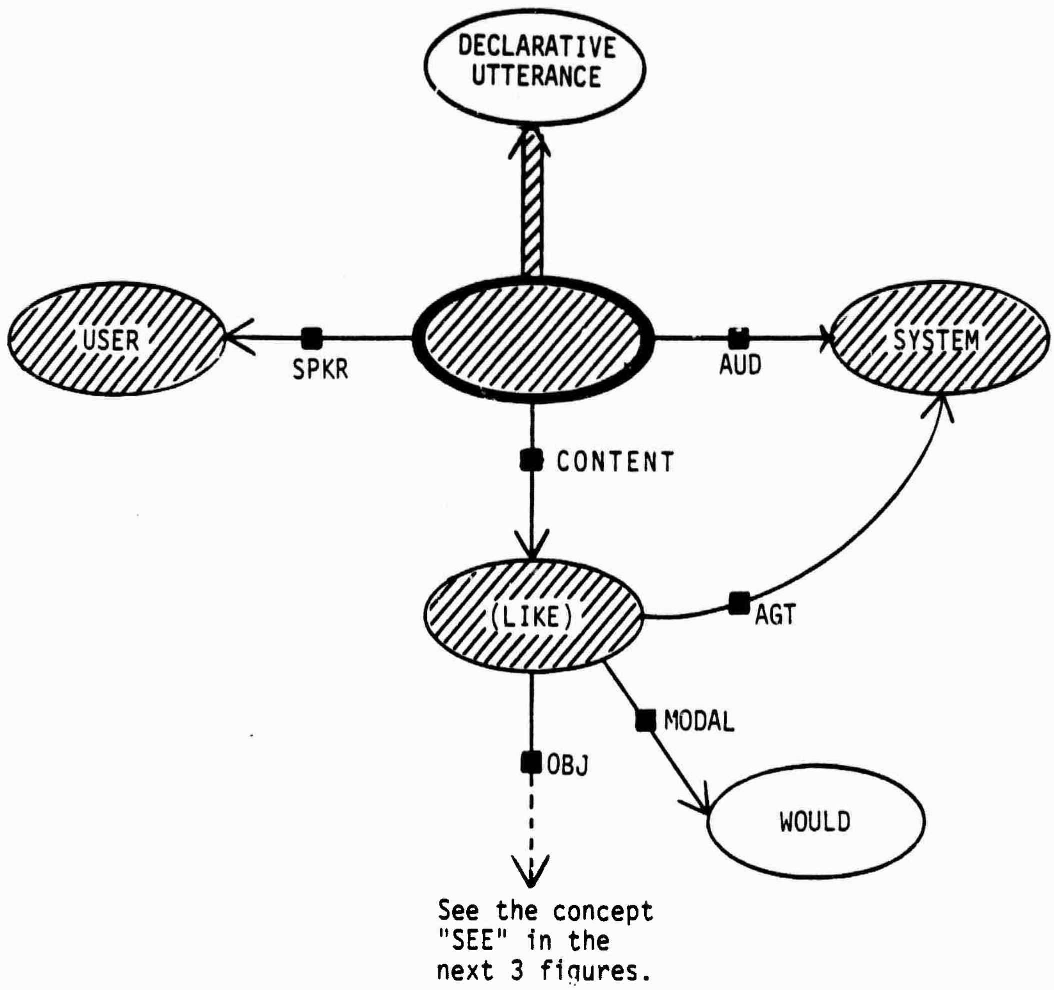


FIGURE 11. I'D LIKE TO SEE THE STRUCTURE BELOW EMPLOYEE BENEFITS.

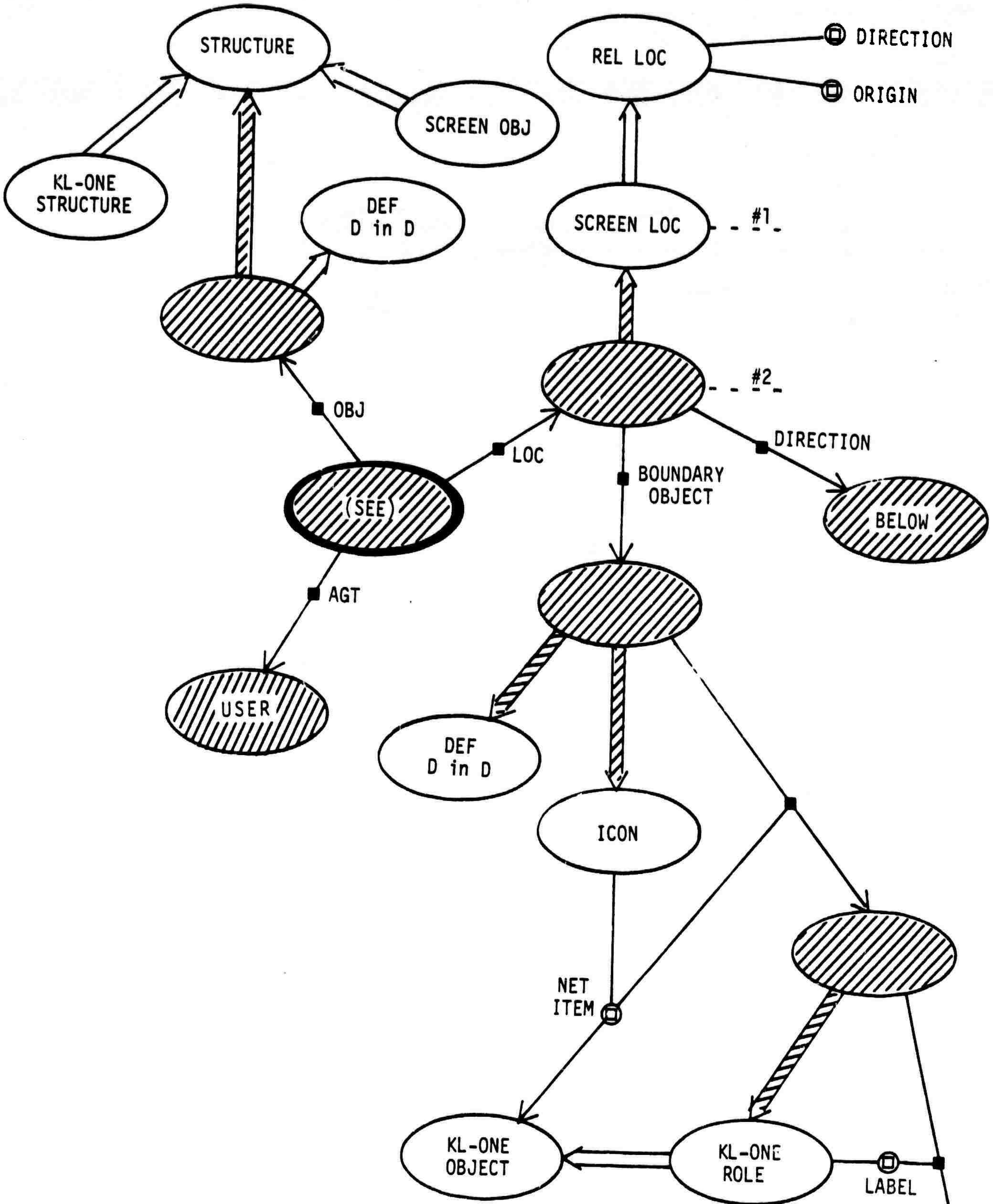


FIGURE 12.
I1: I'D LIKE TO SEE (THE STRUCTURE) (BELOW 'EMPLOYEE BENEFITS')

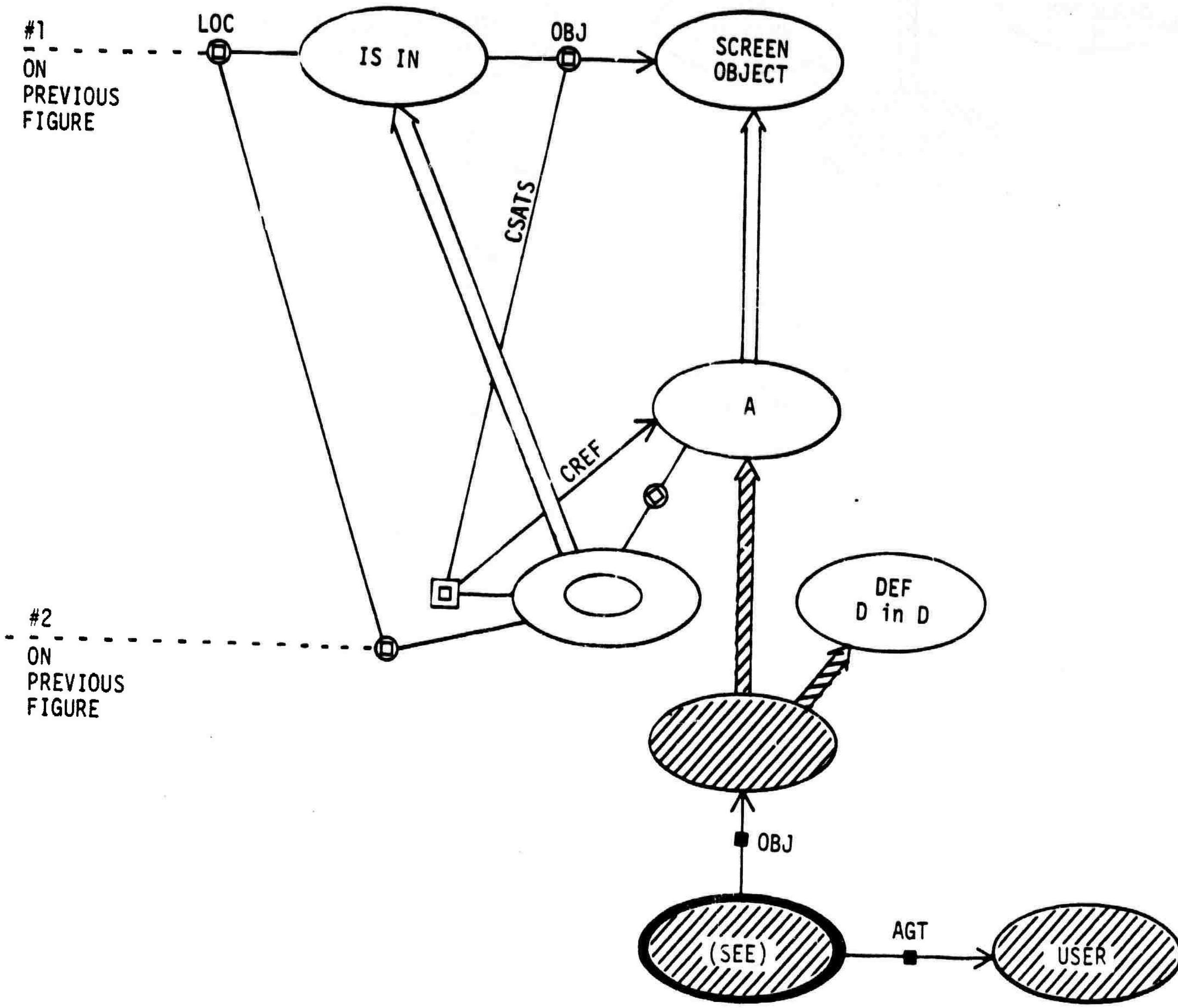


FIGURE 13.
I2: I'D LIKE TO SEE THE (SCREEN) STRUCTURE (THAT IS) BELOW (IN A DIRECTIONAL SENSE) EMPLOYEE BENEFITS.

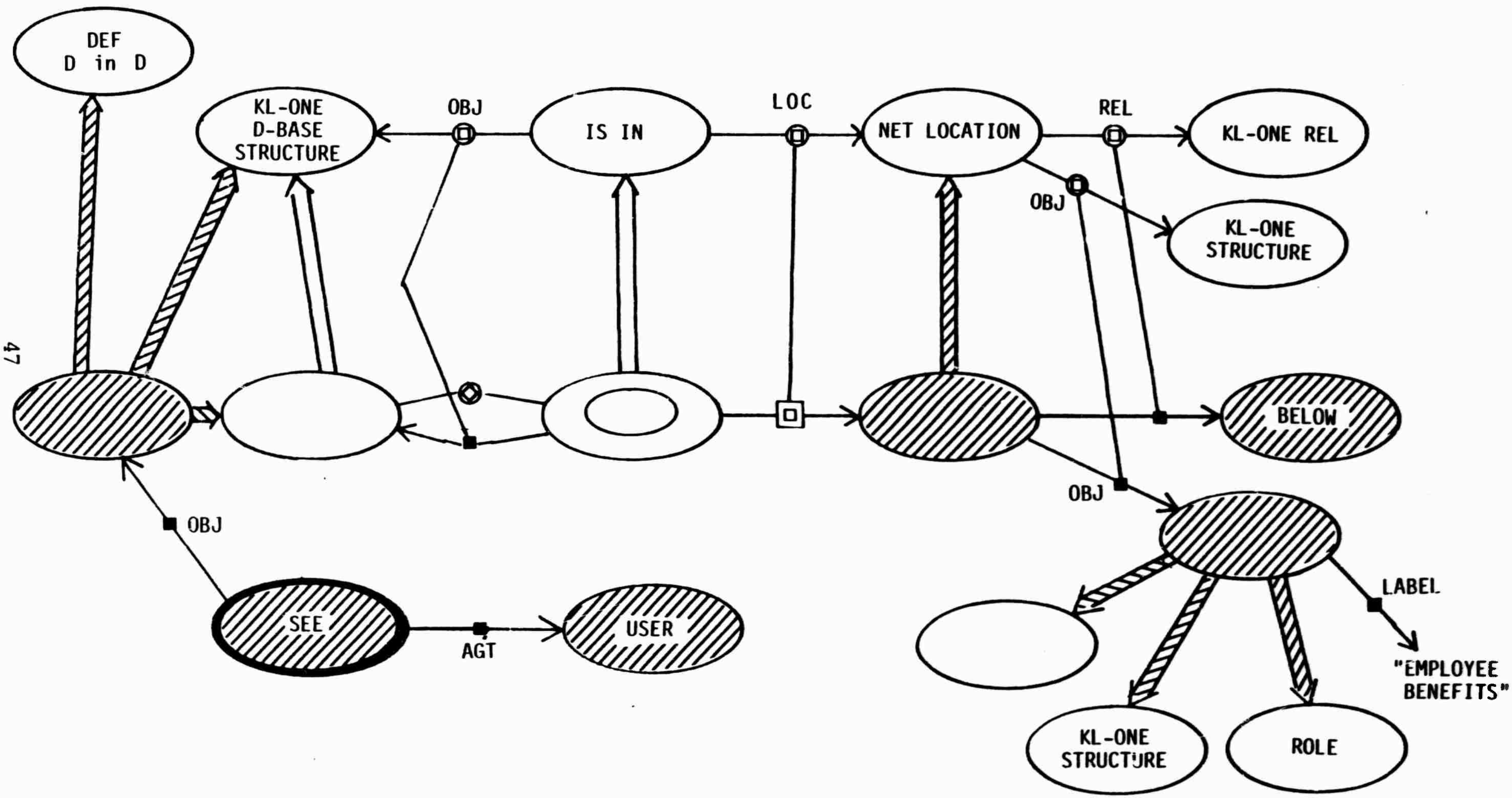


FIGURE 14.
I3: I'D LIKE TO SEE THE (DB) STRUCTURE (THAT IS) BELOW (IN A DIRECTIONAL SENSE) EMPLOYEE BENEFITS.

REFERENCES

[1] Allen, J.F. A plan-based approach to speech act recognition. Technical Report 131, Department of Computer Science, University of Toronto, January, 1979.

[2] Barwise,J. and Perry, J. The Situation Underground. August, 1980.Unpublished manuscript, Stanford University.

[3] Bobrow, R.J. and Webber, B.L. PSI-KLONE - Parsing and Semantic Interpretation in the BBN Natural Language Understanding System. In CSCSI/CSEIO Annual Conference. CSCSI/CSEIO, 1980.

[4] Cohen, P.R. On knowing what to say: Planning speech acts. PhD thesis, University of Toronto, January, 1978. Technical Report No. 118, Dept. of Computer Science.

[5] Grosz, B. J. The representation and use of focus in dialogue understanding. Technical Report 151, Artificial Intelligence Center, SRI International, July, 1977.

[6] Perrault, C.R. and Cohen, P.R. It's for your own good: A note on inaccurate reference. In Joshi, A., Sag, I., & Webber, B. (editors), Elements of Discourse Understanding, . Cambridge University Press, Cambridge, Mass., 1981.

[7] Reichman, R. Plain-speaking: A theory and grammar of spontaneous discourse. PhD thesis, Department of Computer Science, Harvard University, 1981.

[8] Sidner, C.L. Towards a Computational Theory of Definite Anaphora Comprehension in English Discourse. Technical Report 537, Artificial Intelligence Laboratory, Massachusetts Institute of Technology, June, 1979. PhD. Thesis.

[9] Sidner, C.L., Bates, M., Bobrow, R.J., Brachman, R.J., Cohen, P.R., Israel, D., Schmolze, J., Webber, B.L. and Woods, W.A. Research in Knowledge Representation for Natural Language Understanding, Annual Report: 1 September 1980 - 3 August 1981. BBN Report 4785, Bolt Beranek and Newman Inc., 1981.

[10] Sidner, C.L., and Israel, D.J. Recognizing intended meaning and speaker's plans. In Proceedings of the International Joint Conference in Artificial Intelligence, pages 203-208.

The International Joint Conferences on Artifical Intelligence, The International Joint Conferences on Artifical Intelligence, Vancouver, B.C., August, 1981.

[11] Sidner, C.L. Protocols of Users Manipulating Visually Presented Information with Natural Language. Technical Report 5128, Bolt Beranek and Newman Inc., September, 1982.

[12] Sidner, C.L. What the Speaker Means: The Recognition of Speakers' Plans in Discourse. Journal of Computers and Math, forthcoming.

[13] Webber, B.L. A formal approach to discourse anaphora. BBN Report 3761, Bolt Beranek and Newman, May, 1978.

Official Distribution List

Contract N00014-77-C-0378

	Copies
Defense Documentation Center Cameron Station Alexandria, VA 22314	12
Office of Naval Research Information Systems Program Code 437 Arlington, VA 22217	2
Office of Naval Research Code 200 Arlington, VA 22217	1
Office of Naval Research Code 455 Arlington, VA 22217	1
Office of Naval Research Code 458 Arlington, VA 22217	1
Office of Naval Research Branch Office, Boston 495 Summer Street Boston, MA 02210	1
Office of Naval Research Branch Office, Chicago 536 South Clark Street Chicago, IL 60605	1
Office of Naval Research Branch Office, Pasadena 1030 East Green Street Pasadena, CA 91106	1
Office of Naval Research New York Area Office 715 Broadway - 5th Floor New York, NY 10003	1

Naval Research Laboratory Technical Information Division Code 2627 Washington, D.C. 20380	6
Naval Ocean Systems Center Advanced Software Technology Division Code 5200 San Diego, CA 92152	1
Dr. A.L. Slafkosky Scientific Advisor Commandant of the Marine Corps (Code RD-1) Washington, D.C. 20380	1
Mr. E.H. Gleissner Naval Ship Research & Dev. Center. Computation & Mathematics Dept. Bethesda, MD 20084	1
Capt. Grace M. Hopper NAICOM/MIS Planning Branch (OP-916D) Office of Chief of Naval Operations Washington, D.C. 20350	1
Mr. Kin B. Thompson NAVDAC 33 Washington Navy Yard Washington, D.C. 20374	1
Advanced Research Projects Agency Information Processing Techniques 1400 Wilson Boulevard Arlington, VA 22209	1
Capt. Richard L. Martin, USN Commanding Officer USS Francis Marion (LPA-249) FPO New York 09501	1
Director National Security Agency Attn: R54, Mr. Page Fort G.G. Meade, MD 20755	1
Director National Security Agency Attn: R54, Mr. Glick Fort G.G. Meade, MD 20755	1

Major James R. Kreer
Chief, Information Sciences
Dept. of the Air Force
Air Force Office of Scientific
  Research
European Office of A rospace
  Research and Development
Box 14
FPO New York 09510
	1

Dr. Martin Epstein National Library of Medicine Bldg. 38A, 8th Floor Lab 8600 Rockville Pike Bethesda, MD 20209	1